

Write a program to accept a number from the user and display the sum of its digits.

Program to find the sum of digits of a number

```
number = input("Enter a number: ")
n = int(number)
```

```
# Initialize sum
sum_of_digits = 0
```

```
# Loop to extract and add digits
```

```
while n > 0:
    sum_of_digits += n % 10 # extract last digit
    n //= 10               # remove last digit
```

```
# Display the result
print("Sum of digits:", sum_of_digits)
```

Write a program to generate the Fibonacci series.

Fibonacci Series using for loop

```
n = int(input("Enter number of terms: "))
```

```
a, b = 0, 1
Print("Fibonacci Series:")
```

```
For i in range(n):
    Print(a, end=" ")
    A, b = b, a + b
```

#Write a recursive function to print the factorial for a given number.

Version 1: Simple Recursive Factorial

```
def rec(n):  
    if n == 1 or n == 0:  
        return 1  
    else:  
        return n * rec(n - 1)  
  
num = int(input("Please enter the number for factorial: "))  
  
if num < 0:  
    print("Factorial does not exist for negative numbers.")  
  
elif num == 0:  
    print("The factorial of 0 is 1.")  
  
else:  
    print("The factorial of", num, "is", rec(num))
```

Version 2: Without the 0 check inside function

```
def rec(n):  
    if n == 1:  
        return n  
    else:  
        return n * rec(n - 1)  
  
num = int(input("Please enter the number for factorial: "))  
  
if num < 0:  
    print("Factorial does not exist for negative numbers.")  
  
elif num == 0:  
    print("The factorial of 0 is 1.")  
  
else:  
    print("The factorial of", num, "is", rec(num))
```

#Write a program that takes two lists and returns True if they have at least one common member.

```
def common_element(list1, list2):
    result = False
    for i in list1:
        for j in list2:
            if i == j:
                result = True
                return result # Return as soon as a match is found
    return result
```

Input for first list

```
lst1 = []
n = int(input("Enter the number of elements in list1: "))
for i in range(n):
    ele = int(input())
    lst1.append(ele)
print("List1 is:", lst1)
```

Input for second list

```
lst2 = []
n = int(input("Enter the number of elements in list2: "))
for i in range(n):
    ele = int(input())
    lst2.append(ele)
print("List2 is:", lst2)
```

Check for common elements

```
print("Do both lists have a common element?", common_element(lst1,
lst2))
```

```
Enter the number of elements in list1: 4
1
2
3
4
List1 is: [1, 2, 3, 4]
Enter the number of elements in list2: 3
5
6
7
List2 is: [5, 6, 7]
Do both lists have a common element? False
```

#a square if squareBase is True, and the radius of a circle when squareBase is False. y is the height of the object. First, use squareBase to distinguish the cases. Use the circleArea and squareArea from the geometry module to calculate the base areas.

Notepad : geometry.py

```
import math
```

```
def squarearea(side):  
    return side * side
```

```
def circlearea(radius):  
    return math.pi * radius * radius
```

Main:

```
import geometry
```

```
def pointyshapevolume(x, y, squareBase):  
    if squareBase:  
        base = geometry.squarearea(x)  
    else:  
        base = geometry.circlearea(x)  
    return y * base / 3.0
```

Example calls

```
#print(dir(geometry))
```

```
print(pointyshapevolume(5, 4, True))
```

```
print(pointyshapevolume(5, 4, False))
```

#. Enter the number from the user and depending on whether the number is even or odd, print out an appropriate message to the user.

```
number = int(input("Enter a number: "))  
mod = number % 2
```

```
if mod == 0:  
    print("This is an Even Number")  
else:  
    print("This is an Odd Number")
```

#Write a program to accept and pass radius to a function that returns area and circumference (using tuple).

```
number = int(input("Enter a number: "))  
mod = number % 2
```

```
if mod == 0:  
    print("This is an Even Number")  
else:  
    print("This is an Odd Number")
```

#Write a program to accept and pass radius to a function that returns area and circumference (using tuple).

```
import math

def calculate_circle(radius):
    area = math.pi * radius ** 2
    circumference = 2 * math.pi * radius
    return (area, circumference) # Returning as a tuple

radius = float(input("Enter the radius of the circle: "))

area, circumference = calculate_circle(radius)

print(f"Area of the circle: {area:.2f}")
print(f"Circumference of the circle: {circumference:.2f}")
```

Write a program to measure program execution time.

```
import time

def sample_task():
    total = 0
    for i in range(1, 1000000):
        total += i
    return total
```

```
# ----- MAIN -----
```

```

def main():
    start_time = time.time() # Start timer

    result = sample_task()    # Run the task

    end_time = time.time()    # End timer

    execution_time = end_time - start_time

    print(f"Result of computation: {result}")
    print(f"Execution time: {execution_time:.6f} seconds")

if __name__ == "__main__":
    main()

```

#Write a program that compares two dates (in DD/MM/YYYY format) and prints which one is earlier.

```

from datetime import datetime

def compare_dates(date1_str, date2_str):
    # Convert string to datetime object
    try:
        date1 = datetime.strptime(date1_str, "%d/%m/%Y")
        date2 = datetime.strptime(date2_str, "%d/%m/%Y")

        if date1 < date2:
            print(f"{date1_str} is earlier than {date2_str}.")
        elif date1 > date2:
            print(f"{date2_str} is earlier than {date1_str}.")
        else:
            print("Both dates are the same.")
    except ValueError:
        print("Invalid date format. Please use DD/MM/YYYY.")

# ----- MAIN -----
def main():
    d1 = input("Enter the first date (DD/MM/YYYY): ")
    d2 = input("Enter the second date (DD/MM/YYYY): ")

    compare_dates(d1, d2)

# Run the main function
if __name__ == "__main__":
    main()

```

Write a program to accept the SGPI from the user and print corresponding grade based on the following:

```
def get_grade(sgpi):
    if 9.00 <= sgpi <= 10.00:
        return "O"
    elif 8.00 <= sgpi < 9.00:
        return "A+"
    elif 7.00 <= sgpi < 8.00:
        return "A"
    elif 6.00 <= sgpi < 7.00:
        return "B+"
    elif 5.50 <= sgpi < 6.00:
        return "B"
    elif 5.00 <= sgpi < 5.50:
        return "C"
    elif 4.00 <= sgpi < 5.00:
        return "P"
    elif sgpi < 4.00:
        return "F"
    else:
        return "Invalid SGPI"

# Accept SGPI input from user
try:
    sgpi = float(input("Enter your SGPI (0.00 - 10.00): "))
    if 0.00 <= sgpi <= 10.00:
        grade = get_grade(sgpi)
        print(f"Your Grade is: {grade}")
    else:
        print("SGPI must be between 0.00 and 10.00")
except ValueError:
    print("Invalid input. Please enter a numeric value.")
```